N88-25754

ROBUST VECTOR QUANTIZATION FOR NOISY CHANNELS

J.R.B. DE MARCA, Pontificia Universidade Catolica do Rio de Janeiro, Brazil, N. FARVARDIN, University of Maryland, USA, N.S. JAYANT, AT&T Bell Labs, USA, Y. SHOHAM, AT&T Bell Labs, USA;

AT&T Bell Laboratories
600 Mountain Ave.
Murray Hill, New Jersey United States

Abstract

The paper briefly discusses techniques for making vector quantizers more tolerant to transmission errors. Two algorithms are presented for obtaining an efficient binary word assignment to the vector quantizer codewords without increasing the transmission rate. It is shown that about 4.5 dB gain over random assignment can be achieved with these algorithms. It is also proposed to reduce the effects of error propagation in vector-predictive quantizers by appropriately constraining the response of the predictive loop. The constrained system is shown to have about 4 dB of SNR gain over an unconstrained system in a noisy channel, with a small loss of clean-channel performance.

1. Introduction

Vector quantization (VQ) is used for data compression and digital transmission in two basic forms: memoryless VQ and memory-based, vector-predictive quantizers (VPQ) [1,2]. Significant advantage over scalar quantization can be achieved with both systems. However, the effects of *noisy* channels on the VQ and VPQ performance are not yet fully understood. In this paper, we briefly describe two techniques for making VQ and VPQ more robust to transmission errors. The first technique is to use an efficient binary index assignment which can be considered to be zero-bit channel coding. Two algorithms will be described for obtaining an efficient index assignment. The first one is *deterministic* in nature [3] and the second one is a *stochastic* algorithm which is based on the concept of *simulated annealing* [4,5]. Other approaches to this problem can be found in [6,7]. The second technique applies to VPQ and is used to reduce the channel error propagation effect by controlling the response of the vector-predictive loop. This is done by appropriately modifying the root locations of the predictive loop. The robustness to channel errors is obtained without a significant loss in the VPQ performance.

2. The Index Assignment Problem

A vector quantizer of dimension M maps a vector-source symbol $\mathbf{x}$ to a codevector $\mathbf{c}_i$ in a codebook $C = \{\mathbf{c}_0, \mathbf{c}_2, \ldots, \mathbf{c}_{N-1}\}$ of size N by minimizing a given distortion measure $d(\mathbf{x},\mathbf{c}_i)$ with respect to i. The size N determines the rate of the quantizer $B = \log_2 N$ in bits/vector. The indices of the codevectors are transmitted over a digital channel in a form of binary words of length B by using a predetermined binary assignment function $b(i)$.

* C-7

Transmission bit errors alter the received indices and erroneous codevectors are sometimes retrieved from the codebook at the receiver side. Let i and j be the correct and erroneous indices, respectively. The channel causes an additional *channel distortion* $d(\mathbf{c}_j,\mathbf{c}_i)$ on top of the *quantization distortion* $d(\mathbf{x},\mathbf{c}_i)$. Let $p(i)$ be the a-priori codevector probability and $p(j \mid i)$ be the transitional probability of receiving $\mathbf{c}_j$ given that $\mathbf{c}_i$ was transmitted. The transitional probability clearly depends on the mapping $b(i)$ used by the quantizer. The average channel distortion is given by

$$D_c = E\{d(\mathbf{c}_i,\mathbf{c}_j)\} = \sum_{i=1}^{N}\sum_{j=1}^{N} p(i)\,p(j \mid i)\,d(\mathbf{c}_i,\mathbf{c}_j) \tag{1}$$

It can be shown that for the squared-error distortion measure $d(\mathbf{x},\mathbf{c}_i) = \|\mathbf{x}-\mathbf{c}_i\|^2$ used in this study and for an *optimal* (clean-channel) codebook C, the overall average distortion is $D = E\{d(\mathbf{x},\mathbf{c}_j)\} = D_q + D_c$ where $D_q = E\{d(\mathbf{x},\mathbf{c}_i)\}$ is the average quantization distortion.

Since there are many (but finite number of) ways of assigning binary words to codevectors, a fundamental question arises as to what is the best assignment $b(i)$ in the sense of minimum overall distortion D or, equivalently, minimum channel distortion D_c. Two algorithms are suggested below for finding such an index assignment function.

3. Deterministic Algorithm for Binary Index Assignment

This algorithm is based on modeling the noisy channel as a memoryless binary symmetric channel with a negligible probability of having more that one bit error in a word. For this model the transitional probabilities are given by

$$p(j \mid i) = \begin{cases} p_e & \text{if } H(\,b(i),b(j)\,) = 1 \\ 0 & \text{if } H(\,b(i),b(j)\,) > 1 \\ 1-p_eB & \text{if } \mathbf{c}_i = \mathbf{c}_j \end{cases} \tag{2}$$

where p_e is the channel bit error probability and $H(.,.)$ is the Hamming distance between two binary words. Assume that $\mathbf{c}_i$ was transmitted and let $S(i)$ be the set of B *nearest codevectors* to $\mathbf{c}_i$ in the sense of the distortion $d(\mathbf{c}_j,\mathbf{c}_i)$. The distortion induced by the channel is the least if the B binary words $\{\, b(j) : \mathbf{c}_j \,\varepsilon\, S(i) \,\}$ satisfy $H(b(j),b(i)) = 1$. In general it is impossible to satisfy this condition for all the codevectors in C. Therefore, the assignment $b(i)$ has to be determined according to some measure of priority. This measure depends on a-priori probability $p(i)$ and the *conditional distortion* (given $\mathbf{c}_i$) $\sum_{j\neq i} p(j \mid i) d(\mathbf{c}_j,\mathbf{c}_i)$ which, due to (2), is proportional to the sum

$$\alpha(\mathbf{c}_i) = \sum_{j \,\mid\, \mathbf{c}_j \varepsilon S(i)} d(\mathbf{c}_j,\mathbf{c}_i) \tag{3}$$

The following empirical function sets a priority order among the codevectors:

$$F(\mathbf{c}_i) = \frac{p(i)}{\alpha^{\beta}(\mathbf{c}_i)} \tag{4}$$

The parameter β determines the relative importance of the conditional distortion and the a-priori probability. β=0.3 was found experimentally to be a good choice for this parameter.

The binary words are assigned to the codevectors in a decreasing order of $F(\mathbf{c}_i)$. For each $\mathbf{c}_i$, the algorithm attempts to assign binary words to all members of $S(i)$ not previously assigned binary words. First, an initial assignment $\{\, b(i), i=0,..,N-1 \,\}$ is found for the largest $F(\mathbf{c}_i)$. Then, the algorithm attempts to improve it by interchanging the binary assignments of all the pairs (i, j), one pair at a time. However, only successful interchanges which reduce the distortion, are accepted. This index perturbation has been found to significantly improve the initial assignment.

The efficiency of the algorithm was tested by comparing the resulting average distortion to that obtained by a random index assignment. The average gain of the proposed algorithm over random assignment was about 2.7 dB and 4.7 dB in terms of signal-to-noise ratio, for code lengths of $B=3$ and $B=6$, respectively. This indicates a trend of higher efficiency for longer codes which is particularly important for high-dimensional, low-rate coding systems. More details on these experimental results can be found in [3].

4. Stochastic Algorithm Based on Simulated Annealing

Simulated Annealing has successfully been applied to complex combinatorial optimization problems with many local optima. The process mimics an objective function associated with a combinatorial optimization problem as the *energy* associated with a physical system, and by slowly reducing an appropriately defined *effective temperature* of the system, seeks the minimum energy state. Here, the objective function to be minimized with respect to the mapping $b(i)$ is the channel distortion D_c, which depends on $b(i)$ through the transitional probabilities. The mapping vector $b = \{b(i),\ i=0,..,N-1\}$ defines the *state* of the system and the channel error D_c defines the *energy* of the system.

The simulated annealing algorithm works as follows. The effective temperature T is set to an initial high value $T = T_0$ and an initial state b is chosen randomly. Then, a new state b' is obtained by interchanging the assignments of a randomly chosen pair of indices and the change in the distortion (energy) $\Delta D_c = D_c' - D_c$ is calculated. D_c' is the energy associated with the state b'. If $\Delta D_c < 0$, b is replaced by b'. Otherwise, b is replaced by b' with probability $\exp\{-\frac{\Delta D_c}{T}\}$. If the number of energy *drops* exceeds a prescribed number or if too many unsuccessful interchanges (not resulting in energy drops) occur, then, the temperature is lowered. Otherwise, the interchanging continues as above. If the temperature T goes below some prescribed freezing temperature T_f or if it appears that a stable state is reached, the algorithm stops with the final assignment b.

Note that the algorithm allows for "bad" perturbations to occur with some probability in order to climb out from local minima. This probability, however, diminishes as the temperature goes down. It has been shown that with a suitable perturbation scheme and with a sufficiently slow cooling schedule, the algorithm converges with probability one to the globally optimal binary assignment.

The algorithm was tested with a first-order autoregressive Gaussian source to verify its efficiency. 6-dimensional 1 bit/sample optimal vector quantizers were designed for a memoryless Gaussian source with a correlation factor $\rho = 0.0$ and for a heavily correlated Gaussian source with $\rho = 0.9$. To simplify the numerical computations, a single-error-per-word channel was assumed, as for the deterministic algorithm (Sec. 3.), with the transitional probabilities given in (2). In this case, the *normalized* channel distortion is given by

$$\frac{D_C}{p_e} = \frac{1}{M} \sum_{i=0}^{N-1} p(i) \sum_{j:H(b(\mathbf{c}_j),b(\mathbf{c}_i))=1} d(\mathbf{c}_i, \mathbf{c}_j), \tag{5}$$

The algorithm was run with an initial (melting) temperature $T_0 = 10.0$ and a final (freezing) temperature $T_f = 2.5\times10^{-4}$. The *cooling schedule* was as follows. The temperature was reduced by a factor $\alpha = 0.97$ after *five* drops (not necessarily in a row) in the average distortion or after more than 200 perturbations. The algorithm terminated when the temperature dropped below T_f or if 50,000 consecutive perturbations did not result in a drop in the average distortion. The above choices of parameters were obtained experimentally and yielded satisfactory results. Plots A and B of Figures 1 illustrate the evolution of the simulated annealing algorithm for the memoryless and the correlated Gaussian source, respectively, for the dimension $M = 6$. The normalized channel error is plotted as a function of the effective temperature. The figure shows that the algorithm reduced the channel error by about 4.5 dB and 2.0 dB for the memoryless and correlated sources, respectively, compared to random-assignment performance.

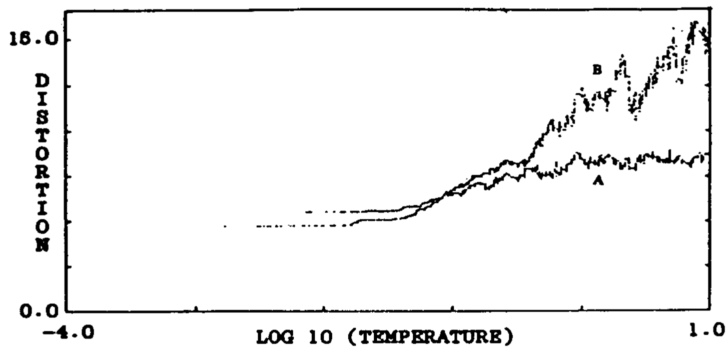

Figure 1. Average channel distortion as a function of the temperature during a run of the Simulated Annealing algorithm. A. Memoryless source (ρ=0.0). B. Correlated source (ρ=0.9).

5. Error Propagation Control in VPQ

In this section we deal with a different type of problem which arises in noisy channels, namely, the problem of error propagation in memory-based coding systems. Specifically, we present a simple technique for reducing such an effect in Vector Predictive Quantizers (VPQ). VPQ is basically an adaptive DPCM type of coding system extended to the vector space [2]. In the VPQ system studied here, an M-dimensional (column) input vector $\mathbf{x}_n$ at a time instant n is quantized in two steps. First, the current vector is predicted from the p immediate past quantized vectors using the M-by-p matrix $\hat{Q}_n = [\hat{\mathbf{x}}_{n-1}, \hat{\mathbf{x}}_{n-2}, \ldots, \hat{\mathbf{x}}_{n-p}]$. The prediction at time n is given by $\tilde{\mathbf{x}}_n = \hat{Q}_n \mathbf{y}$ where $\mathbf{y}$ is a p-dimensional vector of prediction coefficients selected from a *predictor codebook* so as to minimizes the distortion $d(\mathbf{x}_n, \tilde{\mathbf{x}}_n) = \| \mathbf{x}_n - \tilde{\mathbf{x}}_n \|^2$. In the second step, the residual vector $\mathbf{r}_n = \mathbf{x}_n - \hat{Q}_n \mathbf{y}$ is vector-quantized using a second codebook called the *residual codebook*. The quantized error vector $\hat{\mathbf{r}}_n$ minimizes the norm $\| \mathbf{r}_n - \hat{\mathbf{r}}_n \|$. The final quantized output is given by $\hat{\mathbf{x}}_n = \tilde{\mathbf{x}}_n + \hat{\mathbf{r}}_n$. The overall coding rate in number of bits per vector is $B = \log_2 (N_c N_r)$ where N_c and N_r are the predictor and residual codebook sizes, respectively.

This VPQ system is characterized by a time-variant IIR linear system (feedback loop)

$$L_n(z) = \frac{1}{1 - \sum_{i=1}^{p} y_{n,i} z^{-i}} \tag{6}$$

where $y_{n,i}$ is the i^{th} component of $\mathbf{y}$ selected at time n so as to minimize the prediction error. Minimizing the prediction error sometimes yields marginally stable or even unstable system. Thus, $L_n(z)$ may have an extremely long impulse response which tends to spread channel errors over many time frames.

One method for restricting the impulse response duration of the predictive loop is to move the filter poles away from the unit circle. First an unconstrained predictor codebook is designed so as to maximize the prediction gain. Then, the codebook is modified in order to constrain the system impulse response. Let $h(n)$ be an impulse response corresponding to some codevector in the predictor codebook. We denote the effective duration of $h(n)$ by L_e. This quantity is defined by

$$L_e = \max(L) \quad : \quad \frac{\sum_{n=0}^{L} h^2(n)}{\sum_{n=L+1}^{\infty} h^2(n)} \leq T_e \tag{7}$$

where T_e is an effective threshold that is equal to the power ratio of the leading part (0 to L_e) to the tailing part (L_e+1 to ∞) of $h(n)$. The predictor codebook is modified such that the effective lengths of all of its entries are bounded by a predetermined effective length corresponding to a predetermined effective threshold. The modification is achieved by moving the roots of the predictors radially towards the origin, that is, by modifying the predictor coefficients as follows:

$$y_i' = \gamma^i y_i \quad ; \quad 0 < \gamma < 1 \tag{8}$$

The modification expressed in (8) is repeated with some predetermined γ and threshold T_e until (7) is satisfied for a desired length L_e.

This technique was incorporated in vector-predictive quantization of the LPC spectral vectors derived from a speech database at a rate of one per 20 msec. A 20-bit VPQ was designed, with a 6-bit predictor codebook and with a 14-bit residual codebook (which was split into two 7-bit codebooks). The predictor codebook was modified to constrain the loop response by using γ=0.98, T_e=0.01 and L_e=10. Figure 2. shows the response of the time-variant loop L_n to a sequence of pulses spaced 100 time units (frames) apart. Plots A and B show the response of an unconstrained and a constrained loops, respectively, for a VPQ prediction order p=8. While the unconstrained responses may extend over more than 100 frames, the effective length of the constrained responses is around 10 frames.

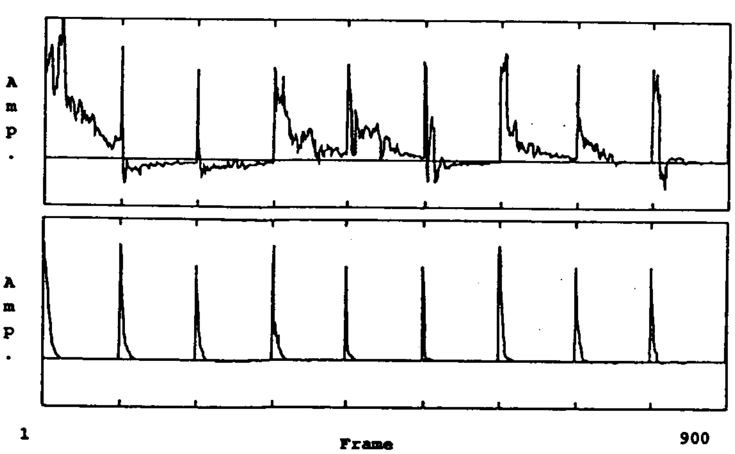

Figure 2. Impulse responses of the time-variant Predictive loop
A. Unconstrained loop. B. Constrained loop.

The VPQ was exposed to channel errors with various bit error rates (BER) and its performance was measured in terms of spectral SNR. The performance figures, summarized in Table 1., show that the constrained VPQ is much more robust to channel errors than the unconstrained quantizer, with a very small degradation of clean-channel performance.

BER	Unconstrained VPQ	Constrained VPQ
0.000	15.44	15.33
0.001	13.94	14.63
0.010	7.57	11.44

Table 1. Spectral SNR (dB) of unconstrained and constrained VPQ's in presence of channel errors.

6. Conclusions

The paper briefly discussed a few techniques for making vector quantizers more tolerant to transmission errors. Two algorithms were presented for an efficient binary word assignment. It was shown that a gain of about 4.5 dB over random assignment could be achieved with those algorithms. A method for constraining the error propagation in predictive VQ was presented. The constrained VPQ was shown to be more robust to channel errors, with a small loss of clean-channel performance.

REFERENCES

1. R. M. Gray, "Vector Quantization," *IEEE Acoust., Speech, Signal Processing Magazine*, Vol. 1, pp. 4-29, April 1984.
2. Y. Shoham, "Vector Predictive Quantization of the Spectral Parameters for Low Rate Speech Coding," *IEEE Proc. ICASSP-87*, Vol. 3, pp. 51.2.1-4, Apr. 1987.
3. J. R. B. DeMarca and N. S. Jayant, "An Algorithm for Assigning Binary Indices to the Codevectors of A Multidimensional Quantizer," *Proc. IEEE Int. Comm. Conf.*, Seattle, WA, pp. 1128-1132, June 1987.
4. S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, Vol. 220, pp. 671-680, May 1983.
5. B. Hajek, "A tutorial Survey of Theory and Application of Simulated Annealing," *Proc. 24th Conf. Dec. & Cont.*, FL, Dec. 85, pp. 755-760.
6. J.-H. Chen, G. Davidson, A. Gersho and K. Zeger, "Speech Coding for the Mobile Satellite Experiment," *Proc. IEEE Int. Comm. Conf.*, Seattle, WA, pp. 756-763, June 1987.
7. K. A. Zeger and A. Gersho, "Zero Redundancy Channel Coding in Vector Quantization," *IEE Electronics Letters*, Vol. 23, pp. 654-655, June 1987.